

WHAT IS CLAIMED IS:

- 1           1. A method for managing utilization of a
- 2      unidirectional stack, comprising the steps:
- 3           initializing a fixed stack marker and a stack base
- 4      for said unidirectional stack;
- 5           upon fetching a program instruction to be executed
- 6      in a computing environment, determining if said program
- 7      instruction involves accessing a location in said
- 8      unidirectional stack;
- 9           if so, further determining whether said location to
- 10     be accessed is within a valid stack range; and
- 11           providing a warning upon determining that said
- 12     location to be accessed is not within said valid stack range.

1           2. The method for managing utilization of a  
2 unidirectional stack as set forth in claim 1, wherein said  
3 computing environment comprises an architectural simulator  
4 operable to simulate a target hardware platform.

1           3. The method for managing utilization of a  
2 unidirectional stack as set forth in claim 2, wherein said  
3 target hardware platform is selected from the group  
4 consisting of a symmetric multiprocessing system, an  
5 asymmetric multiprocessing system, a loosely-coupled  
6 multiprocessing system, and a tightly-coupled multiprocessing  
7 system.

1           4. The method for managing utilization of a  
2 unidirectional stack as set forth in claim 1, wherein said  
3 valid stack range is defined by said stack base and a current  
4 valid stack pointer associated with said unidirectional  
5 stack.

1           5. The method for managing utilization of a  
2 unidirectional stack as set forth in claim 4, wherein said  
3 valid stack range includes said valid stack pointer's  
4 location.

1           6. The method for managing utilization of a  
2 unidirectional stack as set forth in claim 1, further  
3 comprising the step of returning control to a user upon  
4 determining that said location to be accessed is not within  
5 said valid stack range.

1           7. The method for managing utilization of a  
2     unidirectional stack as set forth in claim 1, wherein said  
3     program instruction is operable to perform a read access with  
4     respect to said unidirectional stack.

1           8. The method for managing utilization of a  
2     unidirectional stack as set forth in claim 1, wherein said  
3     program instruction is operable to perform a write access  
4     with respect to said unidirectional stack.

1           9. The method for managing utilization of a  
2     unidirectional stack as set forth in claim 1, further  
3     comprising the step of returning control to an interrupt  
4     handler upon determining that said location to be accessed is  
5     not within said valid stack range.

1           10. The method for managing utilization of a  
2     unidirectional stack as set forth in claim 1, further  
3     comprising the step of specifying a direction of growth for  
4     said unidirectional stack.

1 11. A system for managing utilization of a  
2 unidirectional stack, comprising:

3 means to initialize a fixed stack marker and a  
4 stack base for said unidirectional stack in a computing  
5 environment;

6 means for determining if a program instruction  
7 involves accessing a location in said unidirectional stack,  
8 said program instruction being operable to be executed in  
9 said computing environment; and

10 means for providing a warning upon determining that  
11 said location to be accessed is not within a valid stack  
12 range associated with said unidirectional stack.

1           12. The system for managing utilization of a  
2      unidirectional stack as set forth in claim 11, further  
3      comprising means for returning control to a user upon  
4      determining that said location to be accessed is not within  
5      a valid stack range associated with said unidirectional  
6      stack.

1           13. The system for managing utilization of a  
2      unidirectional stack as set forth in claim 11, wherein said  
3      valid stack range is defined by said stack base and a current  
4      valid stack pointer associated with said unidirectional  
5      stack.

1           14. The system for managing utilization of a  
2      unidirectional stack as set forth in claim 13, wherein said  
3      valid stack range includes said current valid stack pointer's  
4      location.

1           15. The system for managing utilization of a  
2      unidirectional stack as set forth in claim 11, wherein said  
3      computing environment comprises an architectural simulator  
4      operable to simulate a target hardware platform.

1           16. The system for managing utilization of a  
2      unidirectional stack as set forth in claim 15, wherein said  
3      target hardware platform is selected from the group  
4      consisting of a symmetric multiprocessing system, an  
5      asymmetric multiprocessing system, a loosely-coupled  
6      multiprocessing system, and a tightly-coupled multiprocessing  
7      system.

1           17. A method for managing utilization of a  
2 unidirectional stack, comprising the steps:

3                 initializing a fixed stack marker, a stack base and  
4 a stack pointer for said unidirectional stack;

5                 initializing a high water mark for tracking said  
6 stack pointer's location during execution of a program in a  
7 computing environment, said high water mark operating to  
8 identify said stack pointer's farthest location from said  
9 stack base upon completion of said program's execution;

10                upon fetching a program instruction to be executed  
11 in said computing environment, determining if said program  
12 instruction is operable to modify said stack pointer's  
13 current location to a new location in said unidirectional  
14 stack;

15                if so, further determining whether said new  
16 location is within a predetermined stack range; and

17                providing a warning upon determining that said new  
18 location is not within said predetermined stack range.

1           18. The method for managing utilization of a  
2      unidirectional stack as set forth in claim 17, wherein said  
3      predetermined stack range comprises a region bounded by said  
4      stack base and said stack marker.

1           19. The method for managing utilization of a  
2      unidirectional stack as set forth in claim 17, wherein said  
3      predetermined stack range comprises a region bounded by said  
4      stack base and said high water mark.

1           20. The method for managing utilization of a  
2      unidirectional stack as set forth in claim 17, wherein said  
3      computing environment comprises an architectural simulator  
4      operable to simulate a target hardware platform.

1           21. The method for managing utilization of a  
2      unidirectional stack as set forth in claim 20, wherein said  
3      target hardware platform is selected from the group  
4      consisting of a symmetric multiprocessing system, an  
5      asymmetric multiprocessing system, a loosely-coupled  
6      multiprocessing system, and a tightly-coupled multiprocessing  
7      system.

1           22. The method for managing utilization of a  
2      unidirectional stack as set forth in claim 17, further  
3      comprising the step of returning control to a user upon  
4      determining that said new location is not within said  
5      predetermined stack range.



1           23. A system for managing utilization of a  
2 unidirectional stack, comprising:

3               means to initialize a fixed stack marker, a stack  
4 base and a stack pointer for said unidirectional stack;

5               means for tracking said stack pointer's location  
6 during execution of a program in a computing environment,  
7 said means operating to identify said stack pointer's  
8 farthest location from said stack base upon completion of  
9 said program's execution;

10              means for determining if a program instruction is  
11 operable to modify said stack pointer's current location to  
12 a new location in said unidirectional stack; and

13              means for providing a warning upon determining that  
14 said new location is not within a predetermined stack range  
15 associated with said unidirectional stack.

1           24. The system for managing utilization of a  
2   unidirectional stack as set forth in claim 23, wherein said  
3   predetermined stack range comprises a region bounded by said  
4   stack base and said stack marker.

1           25. The system for managing utilization of a  
2   unidirectional stack as set forth in claim 24, wherein said  
3   region includes said stack marker's location.

1           26. The system for managing utilization of a  
2   unidirectional stack as set forth in claim 23, wherein said  
3   computing environment comprises an architectural simulator  
4   operable to simulate a target hardware platform.

1           27. The system for managing utilization of a  
2 unidirectional stack as set forth in claim 26, wherein said  
3 target hardware platform is selected from the group  
4 consisting of a symmetric multiprocessing system, an  
5 asymmetric multiprocessing system, a loosely-coupled  
6 multiprocessing system, and a tightly-coupled multiprocessing  
7 system.

1           28. The system for managing utilization of a  
2 unidirectional stack as set forth in claim 23, wherein said  
3 predetermined stack range comprises a region bounded by said  
4 stack base and a high water mark identified by said means for  
5 tracking said stack pointer's location.

1           29. The system for managing utilization of a  
2 unidirectional stack as set forth in claim 28, wherein said  
3 region includes said high water mark.